Darwin Printing

Session 109

Darwin Printing

Richard Blanchard
Printing

The Search

- Goal
    - Common printing system shared between Darwin and Mac OS X

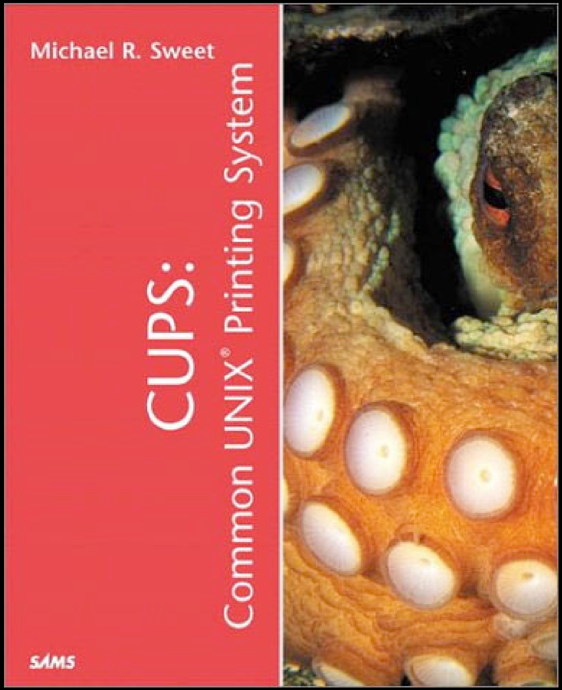
The Search

- Goal
  - Common printing system shared between Darwin and Mac OS X
- Options
  - Write our own
  - Back a strong open source solution

Eureka!

- CUPS
    - Common Unix Printing System
- Available under GPL
    - Suitable for Darwin and Mac OS X
- Strong feature set
- Simple, direct concepts

Well Documented



- Online documentation

  www.cups.org

- The CUPS Book

Layering

- CUPS from **www.cups.org**
  - Includes Xpdf
- CUPS in Darwin
  - Removes Xpdf
  - Xpdf and GhostScript can be added
- CUPS in Mac OS X
  - PDF conversions based on Quartz

Pure CUPS

- Michael Sweet
  - Architect of CUPS from Easy Software Products

An Overview of the Common UNIX Printing System

Michael Sweet
Easy Software Products

What Is CUPS?

- The Common UNIX Printing System (“CUPS”) is a replacement for the old Line Printer Daemon (“LPD”) software based on the Internet Printing Protocol (“IPP”)
- CUPS is a complete *printing solution* for modern printers and applications

Who Uses CUPS?

Besides Apple

- Most Linux and BSD distributions now come with CUPS
- Canon, EPSON, Genicom, OKIDATA, and Xerox all use or support CUPS directly
- The GIMP-print project provides CUPS drivers
- The KDE project uses CUPS to provide enhanced printing capabilities

Why CUPS?

- CUPS is designed to handle modern document formats and printers
    - LPD was only designed to print text files to line printers!
- CUPS provides a consistent printing experience for the user—
    - LPD only handles a limited number of file formats and filters to print those files

Why CUPS?

- CUPS uses IPP and other open standards to provide a better network printing experience
    - LPD only supports the LPD protocol and requires manual configuration of network printers

Why CUPS?

- Adding support for a PostScript printer only involves grabbing the PPD file for the printer
    - PPD files provide access to all the features of a PostScript printer!
- Adding support for a non-PostScript printer involves grabbing a PPD file and one or more filter programs for the printer

Why CUPS?

- The same PPD files can be used on multiple operating systems
- The same non-PostScript printer driver filters programs can be used on multiple operating systems so long as the source code is available

Why IPP?

- IPP was designed for security as well as functionality
  - LPD was, and is, a hack that was developed to support remote printing of text files and does not offer any real level of security
- IPP supports a rich set of standard job template attributes
  - LPD only allows a few specific output filters depending on the type of file you are printing

Why IPP?

- IPP supports authentication, access control, and encryption
    - LPD only provides limited host-based access control

Typical User Experience

- Printing a text file with LPD:

```
lpr filename.txt
lpr –p filename.txt
```

- Printing a text file with CUPS:

```
lpr filename.txt
lpr –p filename.txt
lpr –o prettyprint filename.txt
```

Typical User Experience

- Printing a double-sided PDF document on Legal size paper with LPD:

  How? LPD doesn't support options!

- Printing a double-sided PDF document on Legal size paper with CUPS:

  ```
  lpr –o media=legal –o sides=two-sided-long-edge filename.pdf
  ```

Typical Admin Experience

- Adding a networked PostScript printer with LPD:

```
man 5 printcap
vi /etc/printcap
```

- Adding a networked PostScript printer with CUPS:

```
lpadmin –p name –v device –m ppd-file –E
```

or open the following URL in your web browser:

```
http://localhost:631/admin
```

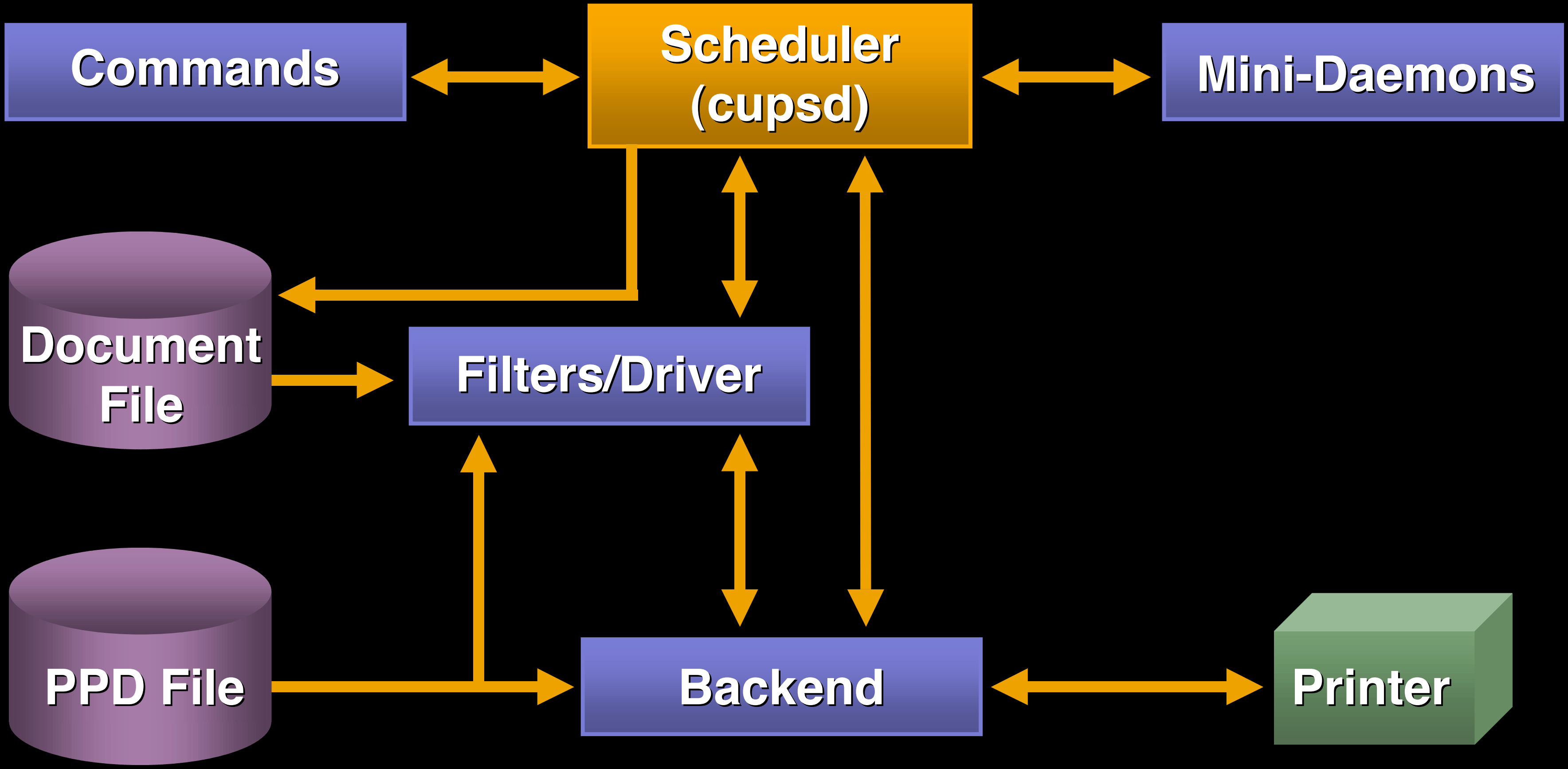
Design Overview

- CUPS implements version 1.1 of the Internet Printing Protocol (“IPP”)
- CUPS uses a small central server that manages queues and jobs
- Applications communicate with the scheduler via IPP to submit print jobs, check queue status, and so forth

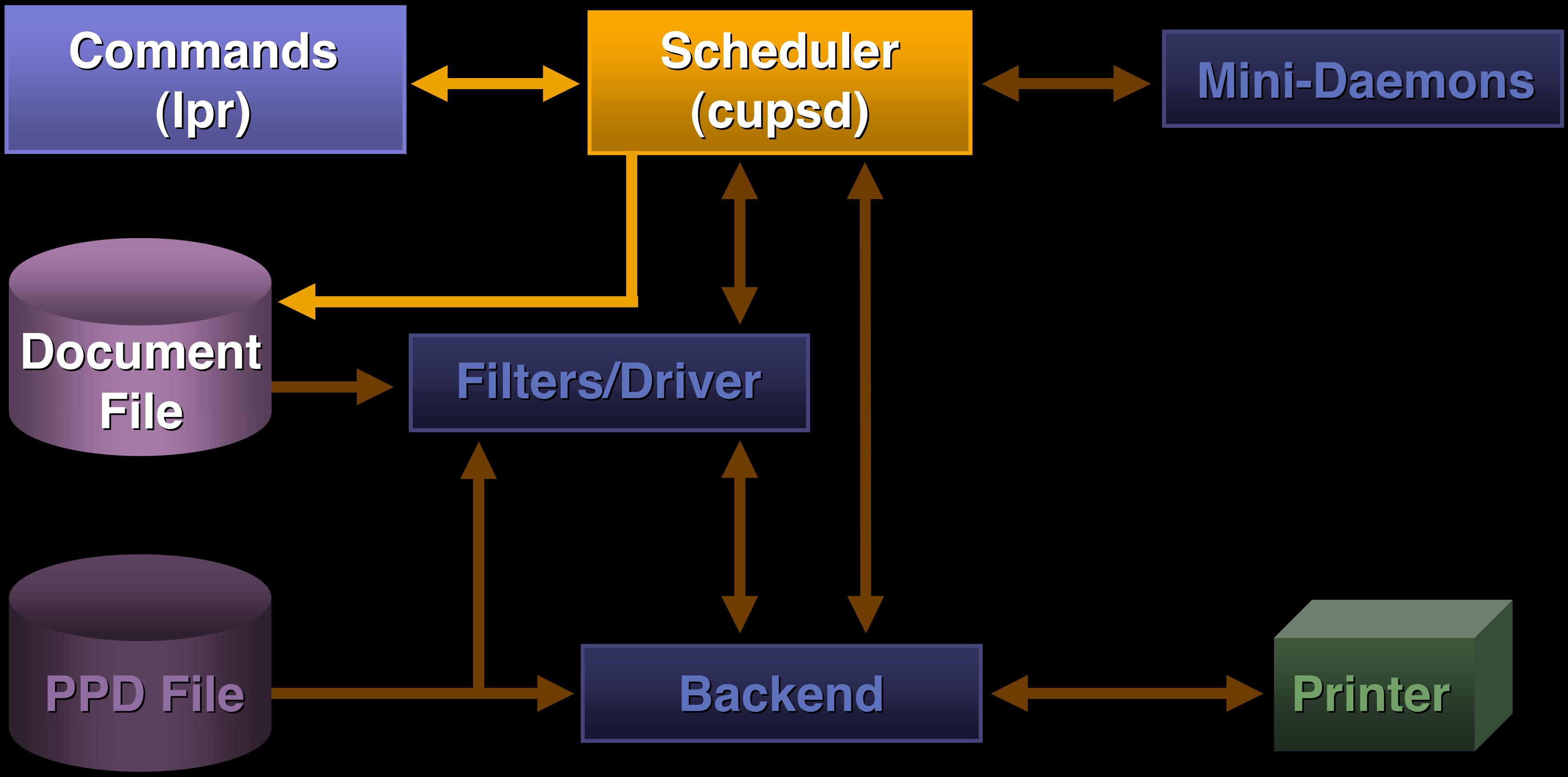
Design Overview

- Users can also access the scheduler via a web browser
- Print jobs are processed by a series of filter programs and sent to a printer using a “backend” program
- Printers, jobs, filters, drivers, and backends are managed dynamically by the scheduler

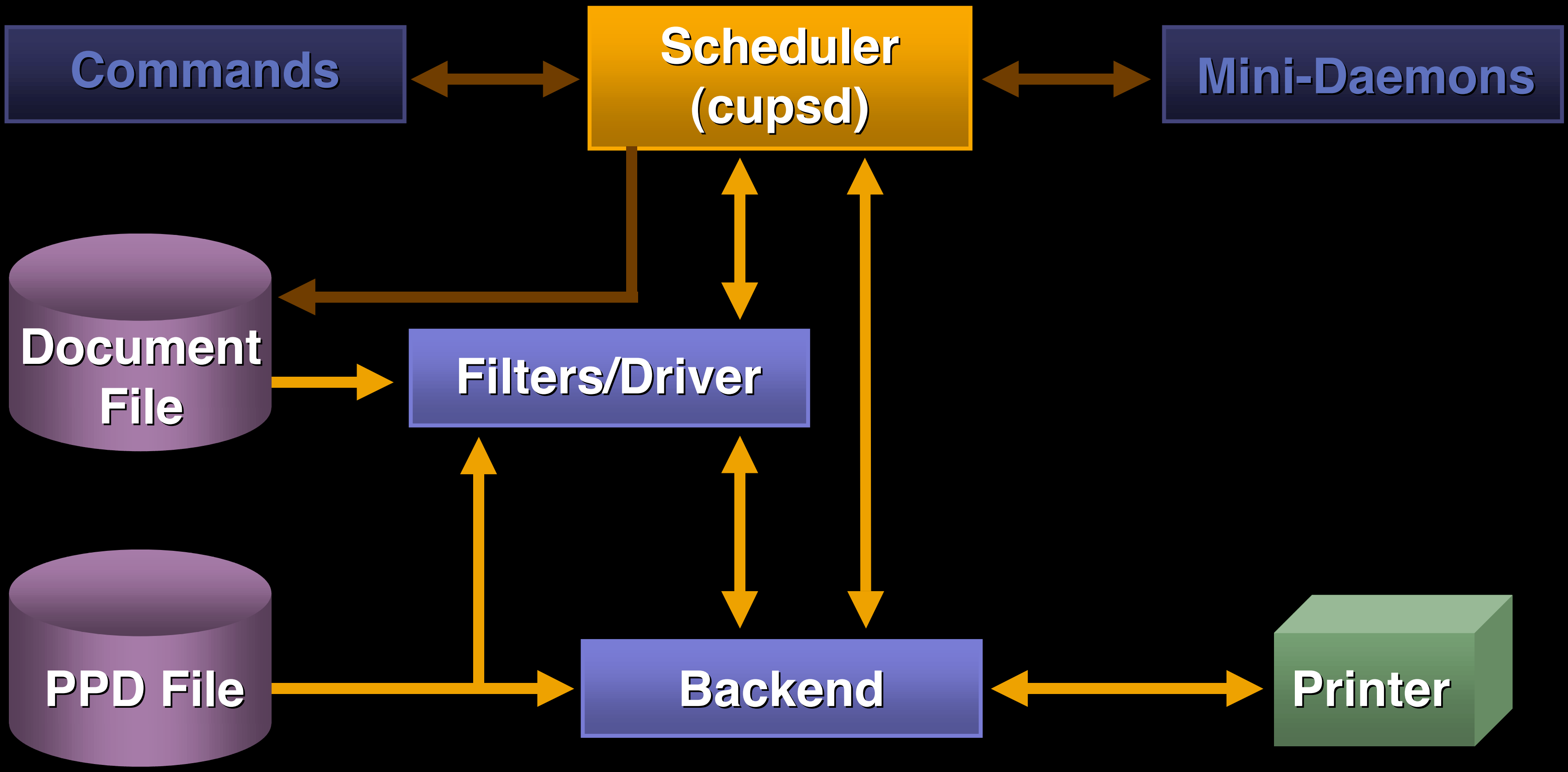
CUPS Block Diagram



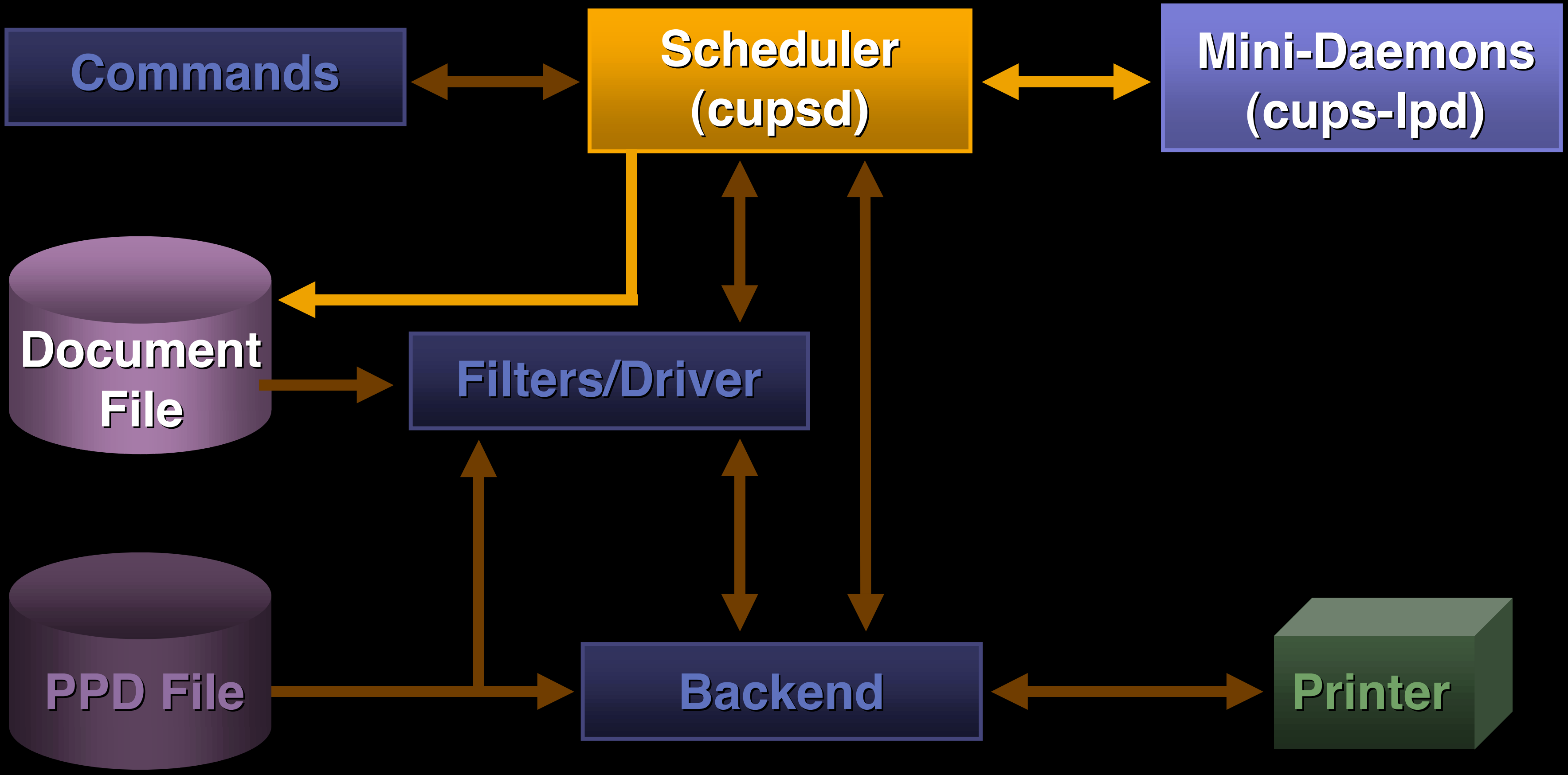
Submitting a File for Printing



Printing a File to the Printer



Supporting LPD Clients



What Does CUPS Provide?

- Berkeley (lpr) and System V (lp) commands
- Web interface
- Extensible filter and backend (device) interfaces
- Job and page accounting/quotas
- Network printer directory services
- IPP and LPD client support
- Encryption support
- PPD-based printer drivers
- Application Programmers Interface and Imaging Libraries
- Documentation

Berkeley Commands

lpc

- Show printer status, etc.

lpq

- Show job status

lpr

- Submit files for printing

lprm

- Cancel jobs

System V Commands

accept, reject

- Accept or reject jobs sent to a queue

cancel

- Cancel print jobs

enable, disable

- Enable (start) or disable (stop) a queue

System V Commands

lp

- Print a file

lpadmin

- Add, modify, and delete queues

lpstat

- Show queue/job status

CUPS Commands

cupsaddsmb

- Export Windows printer drivers to SAMBA

cupsd

- CUPS scheduler/server

CUPS Commands

lpinfo

- Show supported drivers or devices

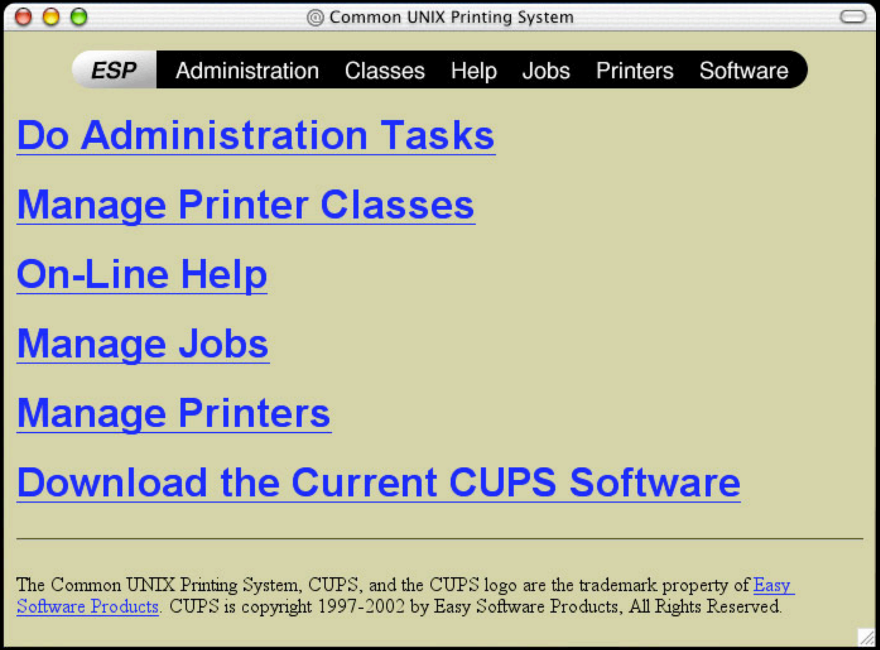
lpmove

- Move print jobs to different queues

lpoptions

- Set and view queue options

Web Interface



Do Administration Tasks

Manage Printer Classes

On-Line Help

Manage Jobs

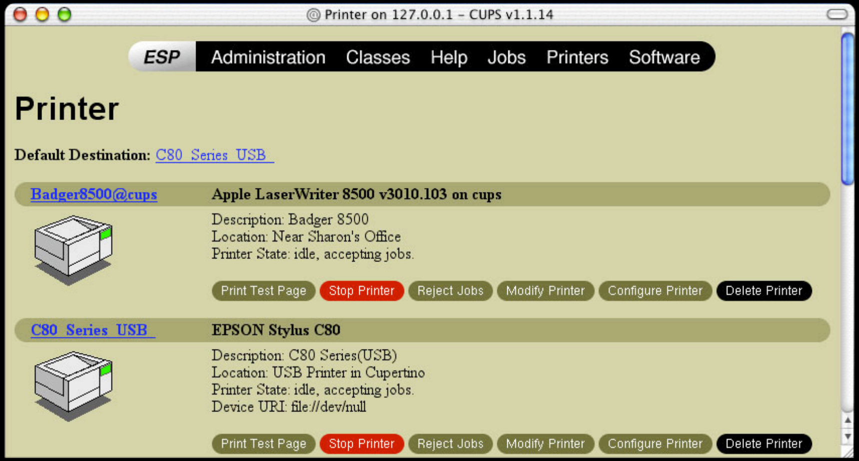
Manage Printers

Download the Current CUPS Software

The Common UNIX Printing System, CUPS, and the CUPS logo are the trademark property of Easy Software Products. CUPS is copyright 1997-2002 by Easy Software Products, All Rights Reserved.

- View documentation
- View job and printer status
- Add, modify, delete, enable, disable, accept, and reject classes and printers
- Configure printer options and features

Web Interface



Printer

Default Destination: C80_Series_USB_

Badger8500@cups **Apple LaserWriter 8500 v3010.103 on cups**

Description: Badger 8500
Location: Near Sharon's Office
Printer State: idle, accepting jobs.

C80_Series_USB_ **EPSON Stylus C80**

Description: C80 Series(USB)
Location: USB Printer in Cupertino
Printer State: idle, accepting jobs.
Device URI: file://dev/null

- Cancel, hold, resume, and restart jobs
- Print a test page
- Customizable via HTML templates
- Localizable

Extensible File Filter Interface

- MIME-based file format and filter registration database
- Scheduler chooses least costly filters to run to convert from the source format to a printable format

Extensible File Filter Interface

- Filters supplied to print international text (8-bit and UTF-8), PostScript, PDF, HP-GL/2, and many types of image files
- Image filters for PostScript and non-PostScript printers

Extensible File Filter Interface

- PostScript RIP filter based on GNU Ghostscript
- Non-PostScript drivers use common raster page stream format to print from PostScript RIP or image file RIP filters; other RIPs can be "dropped in"
- Sample raster driver filters for Dymo, EPSON, Hewlett-Packard, and OKIDATA printers are supplied

Writing a File Filter

- A file filter converts from one format to another
- Some filters are used both as drivers for PostScript printers and filters for non-PostScript printers:
    - Imagetops for printing image files
    - pstops for printing PostScript files

Writing a File Filter

- Most filters produce either PostScript or CUPS raster data
- Filters that produce CUPS raster data are also called RIPs (Raster Image Processors)

Writing a File Filter

- All filters are run with the following arguments:

argv[0] = printer name

argv[1] = job ID

argv[2] = job user name

argv[3] = job name (title)

argv[4] = number of copies

argv[5] = space-delimited options

argv[6] = file to filter (optional)

Writing a File Filter

- All filters are provided with several environment variables:

  CHARSET = character set for the print file

  CONTENT_TYPE = MIME media type for the print file

  CUPS_DATADIR = CUPS data directory

  CUPS_SERVERROOT = CUPS configuration directory, typically **/etc/cups**

Writing a File Filter

- Environment variables (Cont.):

  DEVICE_URI = printer’s device URI

  LANG = the locale name: “en”, “fr”, . . .

  PATH = the standard execution path

  PPD = path to the printer’s PPD file

  PRINTER = the name of the printer queue

Writing a File Filter

- Environment variables (cont.):

  RIP_CACHE = memory limit for RIPs

  SOFTWARE = the name and version number of the server

  TZ = the timezone of the server

  USER = the user executing the filter

Writing a File Filter

- File filters write their output to stdout
- All error/status messages should be written to stderr, prefixed by:
  - "**ERROR:**" = a error that requires attention
  - "**WARNING:**" = a warning message
  - "**INFO:**" = general info ("printing page 1")
  - "**DEBUG:**" = debugging information
- All but debugging messages are passed to the user in the printer-state-message attribute

Writing a File Filter

- File types and filters are registered by adding “foo.types” and “foo.convs” configuration files to **/etc/cups**
- The “foo.types” file lists “magic” rules for identifying a file type:
  application/pdf string(0,%PDF)

Writing a File Filter

- The “foo.convs” file lists filters with the input and output formats and cost:
  application/pdf application/postscript 33 pdftops
- CUPS automatically chooses the lowest cost filtering solution for a particular print job

Writing a Printer Driver

- A printer driver is a specialized file filter
- Printer driver filters are associated with printers through PPD attributes:

```
*cupsFilter: “application/vnd.cups-raster 50 rastertomyprinter”
*cupsFilter: “text/plain 50 texttomyprinter”
```

- Printer driver filters convert from one file format to a “printer native” format

Extensible Backend Interface

- Backends send printable files/data to each printer
- Backends also provide device inventory and discovery (when available) for administration interfaces—this allows the user or a smart client program to pick the correct (available) devices and drivers for each device

Extensible Backend Interface

- Backends can be proxies for other backends
    - An SLP backend could report LPD, AppSocket, or IPP device URIs
- Backends are supplied for parallel, serial, USB, AppSocket, LPD, and IPP connected printers
- SAMBA backend supplied with SAMBA to support printing to Windows servers

Job Accounting Information

- Each job object on the server maintains the job state information
    - Impressions, options, completion times
- 100% accuracy for non-PostScript printers using the supplied drivers
- PostScript printer accounting currently depends on DSC comments

Page Accounting

- A separate **page_log** file can be used to record every page printed on a system; while this does not provide as much information as the job object database, it uses substantially less disk space and memory

Printer Quotas

- Quotas can be set on any printer or class
- Quotas can be based on total print file sizes or pages
- Quotas
    - "100 pages per week"
    - "1000 pages total"
- Currently quotas are enforced the same for all users

Network Printer Directory Services

- CUPS servers provide up-to-date printer information on the LAN using the CUPS or SLPv2 protocols
- Clients automatically configure remote queues
- Clients can automatically create “implicit” classes
- Network printer information can be filtered and/or suppressed via configuration directives

CUPS Browsing Protocol

- The CUPS browsing protocol is a simple broadcast-based protocol
- Servers broadcast small data packets for each printer at regular intervals
    - Typically 80 bytes per printer, every 30 seconds

CUPS Browsing Protocol

- Clients use a timeout mechanism to remove stale remote printers
- Clients and servers can poll and/or relay printer information across subnets

IPP and LPD Client Support

- CUPS is a fully-compliant IPP/1.1 server implementation
- CUPS provides a LPD “mini-daemon” to support LPD clients as needed

Encryption Support

- 128-bit SSL and TLS encryption support
- Most commands support "**-E**" option to encrypt print jobs
- Encryption can be enabled for all requests via client or server configuration options

PPD-based Printer Drivers

- PPDs list printer-specific options:
  - Media sizes, fonts, etc.
- PPDs are used for PostScript and non-PostScript printer drivers
- Non-PostScript drivers use custom PPD attributes to specify:
  - Filters, color profiles, ...

PPD-based Printer Drivers

- PPDs are available via a HTTP GET request
- The CUPS API provides a complete set of functions to load and use PPDs

Application Programmer's Interface

- HTTP functions for accessing HTTP/IPP servers
- IPP functions for managing IPP requests, responses, and attributes
- PPD functions for loading and using PPD files
- Convenience functions for accessing printers, retrieving printer PPD files, sending IPP requests, and managing jobs

Imaging Library

- Functions for colorspace conversion and management
- Functions to manage raster streams (raster page data for printing)
- Image data management functions
- Image file functions
- Nearest-neighbor and bilinear scaling functions

End-User/Developer Documentation

- Software Administrators Manual
- Software Programmers Manual
- Software Users Manual
- Man pages

Distribution Documentation

- Configuration Management Plan
- CUPS Implementation of IPP
- Interface Design Description
- Overview of the Common UNIX Printing System
- Software Design Description
- Software Security Report
- Software Translation Manual
- Software Version Description

CUPS 1.2.x (3Q2002)

- IPv6 support
- ICC color profile support in drivers and RIPs
- Print-URI and Send-URI operation support
- Compressed document support
- IPP notifications
- LDAP support
- Back-channel data path for print drivers

CUPS 1.2.x (3Q2002)

- Firewire (IEEE-1394) backend for Linux and Darwin
- Extended PPD options
- Per-operation and per-printer access policies
- Extended raster data support
  - More user-defined page device attributes and support for 16-bit ICC colorspaces
- Full localization of command-line interfaces

CUPS 1.2.x (3Q2002)

- Transcoding support for all locales
  - E.g., ISO-8859-1 for Canadian French, ISO-8859-15 for French French, all from a common UTF-8 message catalog
- More standard filters and printer drivers
- Device monitor for network, USB, and Firewire printers to handle device connection and disconnection

Internet Resources

- CUPS Website

  http://www.cups.org/

- Easy Software Products Website

  http://www.easysw.com/

- GIMP-print Project (Printer Drivers)

  http://gimp-print.sourceforge.net/

Internet Resources

- KDE Printing Project

  http://printing.kde.org/

- Other CUPS Add-ons

  http://cups.sourceforge.net/

- Printer Working Group

  http://www.pwg.org/ipp/

Specifications

- RFC 2910 and 2911
    - IPP Specifications
- RFC 1179
    - LPD Specification

Roadmap

510 Printing and Mac OS X

Hall 2
Thurs, 10:30am

Who to Contact

Travis Brown
Graphic and Imaging Evangelist
travis@apple.com

Q&A

```
/desktop: No such file or
[duarte:~] ryan% ls
Desktop     Library     Mus
Documents   Movies      Pic
[duarte:~] ryan% cd /Libr
[duarte:/Library] ryan% c
[duarte:/Library/Webserve
CGI-Executables  Document
```

Michael Sweet
Easy Software Products
mike@easysw.com

http://developer.apple.com/wwdc2002/urls.html

WWDC2002

WWDC2002

WWDC2002